PENDING CLAIMS AND STATUS THEREOF

1. **(currently amended):** A method of processing a multiplication operation instruction, comprising:

fetching and decoding a multiplication operation instruction;

executing the multiplication operation instruction on a multiplication source operand to generate a square of the multiplication source operand, **wherein** the multiplication source operand **is** contained in a multiplication operand source register specified in the multiplication operation instruction;

generating a difference output between a minuend source operand and a subtrahend source operand, wherein the minuend source operand and the subtrahend source operand are fetched during execution of a preceding multiplication operation instruction;

storing the difference output in a difference register specified in the multiplication operation instruction; and

storing a result output in a target accumulator specified in the multiplication operation instruction;

wherein the multiplication source operand is transferred directly from the multiplication operand source register to a multiplication execution unit for immediate generation of the square output for the multiplication source operand.

Claims 2-4 (canceled)

5. **(currently amended):** The method according to claim **[[4]] 1**, wherein the multiplication operation instruction specifies a first source register, the first source register

containing an address for the minuend source operand, wherein the minuend source operand is transferred directly from the address to an arithmetic execution unit for immediate generation of the difference output.

6. (original): The method according to claim 5, wherein the multiplication operation instruction specifies a second source register, the source register containing an address for the subtrahend source operand, wherein the subtrahend source operand is transferred directly from the address to an arithmetic execution unit for immediate generation of the difference output.

7. (original): The method according to claim 6, wherein the multiplication operation instruction is a Euclidean Distance operation.

8. (original): The method according to claim 7, wherein the result output is the square of the multiplication source operand.

9. (original): The method according to claim 6, wherein the multiplication operation instruction is an Euclidean Distance Accumulate operation.

10. (original): The method according to claim 9, wherein the step of executing the multiplication operation instruction includes the step of adding the square of the multiplication source operand to an addition source operand, the addition source operand contained in the target accumulator specified in the multiplication operation instruction.

11. (original): The method according to claim 10, wherein the result output is the sum of the square of the multiplication source operand and the addition source operand.

12. (original): The method according to claim 10, wherein the step of executing the multiplication operation instruction includes modifying the address contained in the first source register to contain an address for a next minuend operand for computing a difference during execution of a subsequent multiplication operation instruction.

13. (original): The method according to claim 12, wherein the step of executing the multiplication operation instruction includes modifying the address contained in the second source register to contain an address for a next subtrahend operand for computing the difference during execution of the subsequent multiplication operation instruction.

14. **(currently amended)** The method according to claim 1, further comprising **[[:]]** setting an accumulator status flag

15. **(currently amended)** A processor for multiplication operation instruction processing, comprising:

a program memory for storing instructions including a multiplication operation instruction;

a program counter for identifying current instructions for processing; **[[and]]**

a Digital Signal Processing unit (DSP) for executing instructions within the program memory, the DSP including DSP logic for executing the multiplication operation instruction on a multiplication source operand to generate a square of the multiplication source operand, the multiplication source operand contained in a multiplication source operand register specified in the multiplication operation instruction;

an arithmetic logic unit (ALU) for executing the instructions within the program memory, the ALU including ALU logic for executing the multiplication operation instruction on a minuend source operand and a subtrahend source operand to generate a difference output that is stored in a difference register specified in the multiplication operation instruction, wherein the minuend source operand and the subtrahend source operand are fetched during execution of a preceding multiplication operation instruction; and

a target accumulator for storing a result output, the target accumulator specified in the multiplication operation instruction;

wherein the multiplication source operand is transferred directly from the multiplication operand register to a multiplication execution unit for immediate generation of the square of the multiplication source operand.

Claims 16-18 (canceled)

19. **(currently amended)** The processor according to claim **[[18]]** **15**, further comprising a first Source register containing the address of the minuend source operand, the first source register specified in the multiplication operation instruction, wherein the minuend source operand is transferred directly from the address to an arithmetic execution unit for immediate generation of the difference.

20. (original): The processor according to claim 19, further comprising a second source register containing the address of the subtrahend source operand,. the second source register specified in the multiplication operation instruction, wherein the subtrahend source

operand is transferred directly from the address to an arithmetic execution unit for immediate generation of the difference.

21. (original): The processor according to claim 20, wherein the multiplication operation instruction is an Euclidean Distance operation.

22. (original): The processor according to claim 21, wherein the result output is the square of the multiplication source operand.

23. (original): The processor according to claim 22, wherein the multiplication operation instruction is an Euclidean Distance Accumulate operation.

24. (original): The processor according to claim 23, further comprising the ALU for adding an addition source operand to the square of the multiplication source operand; the addition source operand contained in the target accumulator specified in the multiplication operation instruction;

25. (original): The processor according to claim 24, wherein the result output is the sum of the square of the multiplication source operand and the addition source operand.

26. **(currently amended)** The processor according to claim 15, further comprising **[[the]]** **an arithmetic logic unit (ALU)** **[[ALU]]** for modifying **[[the address]]** **a value** contained in **a** **[[the]]** first source register to **[[contain]]** an address **[[for]]** **of** a **[[next]]** minuend operand **used** for computing a difference during execution of a subsequent multiplication operation instruction.

27. **(currently amended)** The processor according to claim 26, further comprising [[the ALU for]] modifying **[[the address]]** **with the ALU a value** contained in **a** **[[the]]** second source register to **[[contain]]** an address **[[for]]** **of** a **[[next]]** subtrahend operand **used** for computing the difference during execution of the subsequent multiplication operation instruction.

28. **(currently amended)** The processor according to claim 15, further comprising **[[:]]** a status register for indicating accumulator status.